**6.1 매개변수 갱신**

확률적 경사 하강법: 매개변수의 기울기를 구해, 기울어진 방향으로 매개변수 값을 갱신하는 일을 몇 번이고 반복해서 점점 최적의 값에 다가가는 단순한 방법

**6.1.1 모험가 이야기**

모험가는 주위 경치는 볼 수 없지만 지금 서 있는 땅의 기울기는 알 수 있다. 가장 크게 기울어진 방향으로 가자는 것이 SGD 전략이다.

**6.1.2 확률적 경사 하강법**

폰트, 화이트, 텍스트, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

W는 갱신할 가중치 매개변수고, al/aw는 W에 대한 손실 함수의 기울기이다.

<https://github.com/sinyebin/DeepLearning-from-Scratch/blob/master/6.%ED%95%99%EC%8A%B5%EA%B4%80%EB%A0%A8%EA%B8%B0%EC%88%A0%EB%93%A4/SGD.ipynb>

**6.1.3 SGD의 단점**

폰트, 친필, 상징, 화이트이(가) 표시된 사진

자동 생성된 설명

스케치, 그림, 꽃병, 예술이(가) 표시된 사진

자동 생성된 설명

직사각형, 라인, 텍스트, 평행이(가) 표시된 사진

자동 생성된 설명

기울기 대부분은 (0, 0) 방향을 가리키지 않는다.

스케치, 그림, 원, 도표이(가) 표시된 사진

자동 생성된 설명

**6.1.4 모멘텀**

모멘텀: 운동량을 뜻함

폰트, 화이트, 상징, 텍스트이(가) 표시된 사진

자동 생성된 설명

폰트, 화이트, 타이포그래피, 디자인이(가) 표시된 사진

자동 생성된 설명

주방 도구, 포크, 은식기, 주방용품이(가) 표시된 사진

자동 생성된 설명

<https://github.com/sinyebin/DeepLearning-from-Scratch/blob/master/common/optimizer.py>

원, 그림, 도표, 스케치이(가) 표시된 사진

자동 생성된 설명

SGD보다 x축 방향으로 빠르게 다가가 지그재그 움직임이 줄어든다.

**6.1.5 AdaGrad**

학습률 감소: 학습을 진행하면서 학습률을 점차 줄여가는 방법

폰트, 화이트, 타이포그래피, 상징이(가) 표시된 사진

자동 생성된 설명

폰트, 화이트, 타이포그래피, 상징이(가) 표시된 사진

자동 생성된 설명

<https://github.com/sinyebin/DeepLearning-from-Scratch/blob/master/common/optimizer.py>

원, 스크린샷, 텍스트, 도표이(가) 표시된 사진

자동 생성된 설명

최솟값을 향해 효율적으로 움직이는 것을 알 수 있음

**6.1.6 Adam**

모멘텀과 Adagrad를 융합한 것이 Adam

하이퍼파라미터의 편향 보정이 진행됨

원, 그림, 스케치, 도표이(가) 표시된 사진

자동 생성된 설명

**6.1.7 어느 갱신 방법을 이용할 것인가?**

스케치, 도표, 그림, 원이(가) 표시된 사진

자동 생성된 설명

각자의 상황을 고려해 여러 가지로 시도해 보아야 한다.

**6.1.8 MNIST 데이터셋으로 본 갱신 방법 비교**

텍스트, 도표, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

SGD의 학습 진도가 가장 느리고, 나머지 세 기법의 진도는 비슷하다.

하이퍼파라미터인 학습률과 신경망의 구조에 따라 결과가 달라짐

다만 일반적으로 SGD보다 다른 세 기법이 빠르게 학습하고, 때로는 최종 정확도도 높게 나타남

**6.2 가중치의 초깃값**

**6.2.1 초기값을 0으로 하면?**

가중치감소: 가중치 매개변수의 값이 작아지도록 학습하는 방법 가중치 값을 작게 하여 오버피팅이 일어나지 않도록 함

가중치를 작게 만들려면 초기값도 최대한 작게 시작해야된다. 하지만 0으로 하면 학습이 올바르게 이뤄지지 않는다.

초기값이 0일 경우 오차역전파법에서 모든 가중치의 값이 똑같이 갱신되어 모두 같은 값이 전달된다. 가중치가 고르게 되어버리는 상황을 막으려면 초기값을 무작위로 설정해야 한다.

**6.2.2 은닉층의 활성화값 분포**

<https://github.com/sinyebin/DeepLearning-from-Scratch/blob/master/6.%ED%95%99%EC%8A%B5%EA%B4%80%EB%A0%A8%EA%B8%B0%EC%88%A0%EB%93%A4/weight_init_activation_histogram.ipynb>

층이 5개가 있고, 각 층의 뉴런은 100개씩이다.

각 층의 활성화값들이 0과 1에 치우쳐 분포되어 있음

표준편차가 1인 정규분포가 있는데 이는 기울기 소실이라 알려진 문제가 있다.

가중치의 표준평차를 0.01을 설정하면 0.5 부근에 집중, 표현력을 제한한다.

Xavier 초기값을 사용하면 노드가 많을수록 대상 노드의 초기값으로 설정하는 가중치가 좁게 퍼진다.

도표, 라인, 원, 스케치이(가) 표시된 사진

자동 생성된 설명

스케치, 실루엣이(가) 표시된 사진

자동 생성된 설명

**6.2.3 ReLU를 사용할 때의 가중치 초깃값**

He 초깃값은 앞 계층의 노드가 n개일 때, 표준편차 인 정규분포를 사용

도표, 텍스트, 기술 도면, 평행이(가) 표시된 사진

자동 생성된 설명

He 초깃값은 모든 층에서 균일하게 분포되어 있다. 층이 깊어져도 분포가 균일하게 유지되기에 역전파 때도 적절한 값이 나올것으로 기대

**6.2.4 MNIST 데이터셋으로 본 가중치 초깃값 비교**

<https://github.com/sinyebin/DeepLearning-from-Scratch/blob/master/6.%ED%95%99%EC%8A%B5%EA%B4%80%EB%A0%A8%EA%B8%B0%EC%88%A0%EB%93%A4/weight_init_compare.ipynb>

텍스트, 도표, 라인, 그래프이(가) 표시된 사진

자동 생성된 설명

0.01일 때 학습이 전혀 이뤄지지 않다. 순전파 때 너무 작은 값이 흐르기 때문에 가중치가 거의 갱신되지 않는다.

**6.3 배치 정규화**

**6.3.1 배치 정규화 알고리즘**

학습을 빨리 진행할 수 있다.

초깃값에 크게 의존하지 않는다.

오버피팅을 억제한다. 각 층에서의 활성화값이 적당히 분포되도록 조정하여 데이터 분포를 정규화하는 배치 정규화 계층을 신경망에 삽입

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

친필, 폰트, 텍스트, 화이트이(가) 표시된 사진

자동 생성된 설명

라인, 도표, 화이트이(가) 표시된 사진

자동 생성된 설명

**6.3.2 배치 정규화의 효과**

텍스트, 도표, 라인, 그래프이(가) 표시된 사진

자동 생성된 설명

도표, 평면도, 라인, 패턴이(가) 표시된 사진

자동 생성된 설명

배치 정규화를 사용할 때의 학습 진도가 빠르다.

배치 정규화를 이용하지 않는 경우엔 초깃값이 잘 분포되어 있지 않으면 학습이 전혀 진행되지 않는다.

**6.4 바른 학습을 위해**

**6.4.1 오버피팅**

매개변수가 많고 표현력이 높은 모델, 훈련 데이터가 적을 때 일어남

도표, 텍스트, 라인, 그래프이(가) 표시된 사진

자동 생성된 설명

100 에폭을 지나는 무렵부터 거의 100%이다. 그러나 시험 데이터에 대해서는 큰 차이를 보임, 훈련 데이터에만 적응하여 정확도가 크게 벌어짐

**6.4.2 가중치 감소**

큰 가중치에 대해서는 그에 상응하는 큰 페널티를 부과하여 오버피팅을 억제하는 방법

텍스트, 도표, 라인, 그래프이(가) 표시된 사진

자동 생성된 설명

훈련데이터에 대한 정확도와 시험 데이터에 대한 정확도에는 여전히 차이가 있지만, 가중치 감소를 이용하지 않은 것과 비교하면 그 차이가 줄었다.

**6.4.3 드롭아웃**

가중치 감소는 간단하게 구현 가능하고, 어느정도 지나친 학습을 억제할 수 있지만 신경망 모델이 복잡해지면 가중치 감소만으로는 대응하기 어려워져 드롭아웃이라는 기법을 이용한다.

스케치, 도표, 라인, 화이트이(가) 표시된 사진

자동 생성된 설명

도표, 텍스트, 라인, 그래프이(가) 표시된 사진

자동 생성된 설명

드롭아웃을 이용하면 표현력을 높이면서도 오버피팅을 억제 가능

**6.5 적절한 하이퍼파라미터 값 찾기**

**6.5.1 검증 데이터하이퍼파라미터**

하이퍼파라미터 조졍용 데이터를 일반적으로 검증 데이터라고 부름

**6.5.2 하이퍼파라미터 최적화**

무작위로 하이퍼파라미터 값을 골라낸 후, 그 값으로 정확도를 평가하고, 정확도를 잘 살피면서 이 작업을 여러 번 반복하며 하이퍼파라미터의 최적 값의 범위를 좁혀간다.

0단계: 하이퍼파라미터 값의 범위를 설정

1단계: 설정된 범위에서 하이퍼파라미터의 값을 무작위로 추출

2단계: 1단계에서 샘플링한 하이퍼파라미터 값을 사용하여 학습하고, 검증 데이터로 정확도를 평가

3단계: 1단계와 2단계를 특정횟수 반복하여, 그 정확도의 결과를 보고 하이퍼파라미터의 범위를 좁힘

**6.6 정리**

매개변수 갱신 방법에는 확률적 경사 하강법 외에도 모멘텀, AdaGrad, Adam 등이 있다.

가중치 초깃값을 정하는 방법은 올바른 학습을 하는 데 매우 중요

가중치의 초깃값으로는 ‘Xavier 초깃값’과 ‘He 초깃값’이 효과적이다.

배치 정규화를 이용하면 학습을 빠르게 진행할 수 있으며, 초깃값에 영향을 덜 받게 된다.

오버피팅을 억제하는 정규화 기술로는 가중치 감소와 드롭아웃이 있다.

하이퍼파라미터 값 탐색은 최적 값이 존재할 법한 범위를 점차 좁히면서 하는 것이 효과적이다.